

Constraining deformable templates for shape recognition

Teresa Aixut, Yuri L. de Meneses^{*}, Fabien Bourgeois, Jacques Jacot
Laboratoire de Production Microtechnique-Institut de Production et Robotique
Ecole Polytechnique Fédérale de Lausanne (EPFL), CH-1015 Lausanne, Switzerland

ABSTRACT

This paper addresses the problem of robust shape recognition in the presence of shape deformation as well as changes in part position, orientation and scale. Point Distribution Models (PDM) are deformable templates that have interesting features for industrial inspection tasks, since they are built by statistical analysis of a training set and they define a prototype shape as well a set of possible, acceptable deformations. To further improve their classification capabilities, these deformable templates are extended by adding a constraint on the amount of deformation. A constrained optimization procedure is proposed and successfully tested on an industrial inspection task.

Key words: Deformable templates, Point Distribution Models, constrained models, inspection, classification, Hotelling's T^2 , Linear Matrix Inequality, robust shape recognition

1.INTRODUCTION

The objective is to build a system capable of inspecting and classifying parts according to their shape, in the environment of a manufacturing chain. In such a context, images are exposed to translation, rotation, scaling, noise and perspective projection. Additionally to these image distortions, one can also find deformations in the contour of the shape. Sometimes, these deformations fit inside the specifications; other times, they can not be tolerated and the parts must be rejected. Not only the system must be capable of classifying parts into different classes, but it must be able to verify that the different parts conform the given specifications. Some quality measures are done related to the specifications to evaluate the capability of the process.

Computer vision is used in manufacturing chains for many tasks, that range from part recognition to quality assessment. Being able to analyze and classify parts and reject those that do not conform to the minimum specifications is highly desirable in this environment. However, in assembly conveyor lines, a large set of different parts can appear in random orientation and position and an image of the same object can also vary in appearance, depending on lighting and angle. Part recognition in such situations becomes more difficult. The usual requirements in such a context are:

- Reliability: the system must be robust to noise and image variations.
- Speed: the decision should be provided so no delay is introduced in the production chain.
- Automation: limited user interaction.

Many different techniques have been proposed to achieve efficient systems capable of matching these conditions. Some works propose shape correspondence based on shape decomposition, vectorization and feature extraction¹. Neural networks are also used to match images after feature extraction of the objects^{2,3,4}, but despite the good results, they require huge stored data bases or a heavy training. Other strategies make use of segmentation by polygonal shapes⁵ that have the advantage of dealing with shapes including holes. Deformable templates have demonstrated to be effective in the analysis of parts which present variations in shape and appearance^{6,7,8}. Non-linear deformations of the model have also been investigated⁹ but these techniques do not restrict the amount of deformation in a shape.

^{*}. Corresponding author. E-mail: yuri@ieee.org; Phone: +41 21 693 39 08; Fax: +41 21 693 38 91

A Point Distribution Model (PDM) is a type of deformable template that has the advantage of being built by statistical analysis on a training set, rather than making by hand a specific model for each class^{10,11}. Although PDM impose a constraint on the deformation directions, it does not impose a limit in the quantity of deformation in each direction, so a new limit should be found to restrict the range of deformation along these directions. This paper extends the work done in shape analysis using deformable templates^{7,12} by adding a new constraint to improve its classification capabilities. This constraint is based on a statistical control chart widely used in manufacturing process to reject outliers.

This paper will start with an exposition of deformable templates to model shape contours, giving an overview of the elements involved, as well as an explanation of the training method used to obtain the models. In the same section, we will extend the template with a constrained optimization. In the second section, the classification method will be exposed. Afterwards, the results obtained from different tests will be shown, driving us to some conclusions.

2.DEFORMABLE TEMPLATES: POINT DISTRIBUTION MODELS

A template is a model of an image object described by the spatial distribution of its features. Deformable templates based on object contours define the model as a prototype, i.e. a representative contour of the shape, and a set of possible deformations on the template, so each model takes into account all the possible deformations of the object. Several implementations of deformable models have been proposed (snakes, parametric shapes...). We have selected the Point Distribution Models (PDM) because they can be used to capture a priori knowledge of the shape at hand⁷ and they afford a statistical interpretation that is of interest in quality control applications.

Given an object in an image, the set of N points lying on its contour forms a vector

$$u = [u_{x0} \dots u_{xN} \ u_{y0} \dots u_{yN}]. \quad (1)$$

in what we shall call image space.

These points are a scaled, rotated and translated version of a given shape x which “lives” in shape space. Image and shape spaces are related by

$$u = Mx + t, \quad (2)$$

with

$$M = s \begin{bmatrix} \cos \theta I_N & -\sin \theta I_N \\ \sin \theta I_N & \cos \theta I_N \end{bmatrix} \quad t = [t_x \dots t_x \ t_y \dots t_y]^T. \quad (3)$$

PDMs consider that the shape x is a linear combination of a mean shape x_m and a set of L deformations p_i . This can be expressed as

$$x = x_m + Pb \quad P = [p_1 \dots p_L]. \quad (4)$$

And therefore the relationship between image space and shapespace can be rewritten as

$$u = M(x_m + Pb) + t. \quad (5)$$

Thus, the model capable of describing a class of shapes is made up of the pair $\{x_m, P\}$.

2.1 Constructing a PDM: training phase

Models are built by performing a statistical analysis of a training set. The point distribution models starts from a labelled training set, i.e. the shape model must be laid-out on the images in each image of the training set, and the contours in the training set are rotated and scaled so that they can be aligned. This can be performed iteratively, by requesting a human intervention ⁷ or semi-automatically.

Once the training set is aligned, the mean shape is obtained as the average of the coordinates of the shapepoints that lie on the contour of the shape (cf. Figure 1):

$$x_m = \frac{1}{K} \sum_{i=1}^K y_i \quad (6)$$

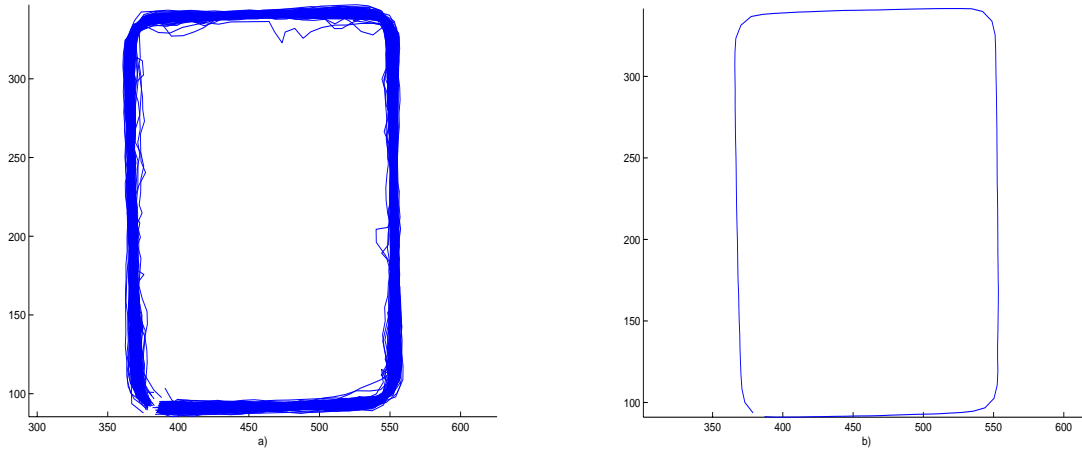


Figure 1: a) Aligned shapes of the training set. b) resulting mean shape.

The modes of variation, the ways in which the points of the shape tend to move together, can be found by applying Principal Component Analysis (PCA)¹³ to the deviations from the mean. The principal components of the deformation are provided by the eigenvectors of the covariance matrix. The covariance matrix S is calculated as:

$$S = \frac{1}{K} \sum_{i=1}^K (y_i - x_m)^T (y_i - x_m) \quad (7)$$

The eigenvectors of the covariance matrix provide a set of orthogonal deformations. The shapepoints are allowed to move in the directions defined by these eigenvectors relative to the mean shape.

The corresponding eigenvalues represent the amount of energy associated to each deformation mode ¹³. The eigenvectors corresponding to the largest eigenvalues describe the most significant modes of variation. Thus the matrix P is formed by the subset of the eigenvectors of the covariance matrix S that accounts for a high percentage of the total energy. Figure 2 shows the three main deformation modes for the shapes in the figure above.

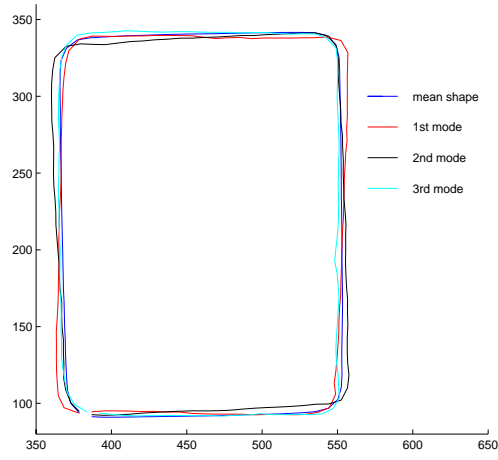


Figure 2: Three main modes of variation with respect to the mean: deformation modes

The ratio of each eigenvalue to the sum of the eigenvalues gives the influence of each mode to the total variation:

$$\zeta(i) = \frac{\lambda_i}{\sum_{i=1}^m \lambda_i} \quad (8)$$

Figure 3 shows the cumulative energy in the deformation modes found in the example shown in Figure 1 and Figure 2. 100 shapepoints have been used to describe the contour, thus producing 100 deformation modes. However, the 13 first deformation modes account for approximately the 90% of the deformations of the training set and the 95% is achieved with the 24 first eigenvectors.

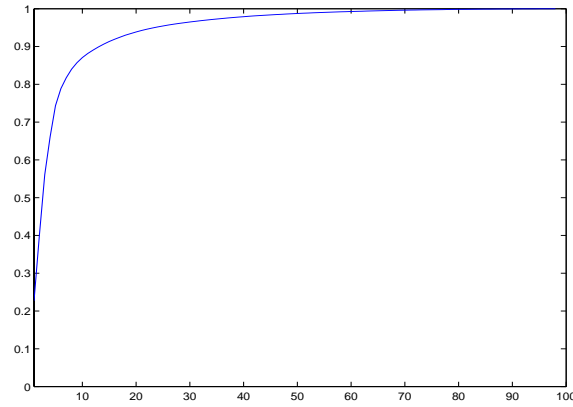


Figure 3: Cumulative energy in the deformation modes.

2.2 Fitting data to a PDM

Once a series of PDM models -one for each class of objects- are built, performing shape recognition consists in determining which model best fits the contour data. The problem to be considered is that of finding the set of values

$\{s, \theta, t_x, t_y, b\}$ (where s gives the scale, θ the rotation, t_x and t_y the translation and b is the set of deformation parameters) that best approximates a given set of points v in image space. These parameters are calculated using least squares measures of the error between the shape v and the shape given by the model $\{x_m, P, b\}$.

First the pose parameters $\{s, \theta, t_x, t_y\}$ are calculated, and then an estimation of the deformation parameters b can be obtained. The error to consider now is given by

$$e_u = v - M^{-1}(x_m + Pb) - t. \quad (9)$$

We express the error in shapespace rather than image space. The resulting expression is

$$e_x = (M^{-1})^T (v - t) - (x_m + Pb). \quad (10)$$

We are looking for the parameter vector b that minimizes the error. This is an optimization problem that can be solved by several optimization techniques⁷.

The Point Distribution Model imposes some constraints on the results while fitting the contour of a shape. As it has been seen, the modes of deformation of the model are obtained via PCA. These modes reflect the main directions of the observed deformations during the training phase, limiting the degrees of freedom of the fitting process.

2.3 Constraining the shape

A class of shapes is not only defined by the deformation modes but also by the amount of acceptable deformation. The possible values of b must be restricted to lie within the limits observed in the training set. The shape approximation problem becomes a constrained approximation problem. With the constrained optimization the deformation modes define the deformation directions and the limits imposed on the parameters b define the range of acceptable deformation along these directions.

Hotelling's T^2 chart is an overall measure of variability of an observation vector with respect to an established standard. This control chart is widely used to detect and remove outliers^{13,14}. Here it is employed to add a constraint to the deformable template during the optimization so the model can not fit a part that does not satisfy the requirements. A confidence interval is defined while the limit is fixed, so this constraint can change depending on the value of the confidence interval. This upper bound value on the deformation can be calculated as¹³

$$T^2 = y^T y, \quad (11)$$

where y is defined as

$$y = x - \bar{x}. \quad (12)$$

From the PDM we have that x can be expressed as:

$$x = \bar{x} + \tilde{P}b \quad (13)$$

where \bar{x} is the mean shape and \tilde{P} is the matrix of the main eigenvalues of the covariance matrix normalized to unit variance, defined as

$$\tilde{P} = P\Lambda^{-1/2}R, \quad (14)$$

where

$$P = [u_1 \dots u_m] \quad R = \begin{bmatrix} 1 & 0 & 0 & \dots & 0 \\ 0 & 1 & 0 & \dots & 0 \\ \cdot & \cdot & \cdot & \dots & \cdot \\ 0 & 0 & 0 & \dots & 1 \\ \cdot & \cdot & \cdot & \dots & \cdot \\ 0 & 0 & 0 & \dots & 0 \end{bmatrix} \quad \Lambda = \begin{bmatrix} \lambda_{max} & 0 & \dots & 0 \\ \cdot & \cdot & \cdot & \cdot \\ 0 & 0 & \dots & \lambda_{min} \end{bmatrix}. \quad (15)$$

The vector u_i represents the eigenvector associated to the i^{th} largest eigenvalue, R is a $m \times n$ matrix used to reduce the number of variables involved in the problem being m the total number of variables and n the number of selected eigenvalues. Λ is a diagonal matrix with the eigenvalues of the covariance matrix. Now T^2 can be rewritten as

$$T^2 = (\tilde{P}b)^T \tilde{P}b = b^T \tilde{P}^T \tilde{P}b = b^T R^T (\Lambda^{-1/2})^T P^T P \Lambda^{-1/2} Rb. \quad (16)$$

Since P is orthogonal, the equation above simplifies to

$$T^2 = b^T R^T \Lambda^{-1} Rb. \quad (17)$$

Any vector b which produces values greater than $T_{\alpha, p, n}^2$ will be out of control on the T^2 chart. That is,

$$T_{\alpha, p, n}^2 \geq b^T R^T \Lambda^{-1} Rb \quad (18)$$

where $T_{\alpha, p, n}^2 = \frac{p(n-1)}{n-p} \cdot F_{p, n-p, \alpha}$ is an univariate distribution proportional to the F distribution with n being the number of observations, p the number of selected variables and α the desired confidence interval¹³.

Multivariate normality is been assumed during the computation of the constraint. When non-normality is detected a generalization of Hotelling's T^2 can be applied¹⁵ to correct its effect; if dispersion in data is small and for large number of samples the generalized T^2 behaves almost exactly the same as the classical T^2 .

The data fitting process is modified accordingly. The minimization of the least square error is constrained according to

$$T^2 \geq b^T R^T \Lambda^{-1} Rb. \quad (19)$$

Applying Schur complements equation (19) can be expressed as the linear matrix inequality¹⁶

$$\begin{bmatrix} T^2 & b^T \\ b & \Lambda \end{bmatrix} \geq 0. \quad (20)$$

Therefore we now have a quadratically constrained quadratic problem like:

$$\text{minimize } e^T e \quad \text{subject to } \begin{bmatrix} T^2 & b^T \\ b & \Lambda \end{bmatrix} \geq 0. \quad (21)$$

where $e^T e$ is the quadratic error previously introduced in equation (10).

3.CLASSIFICATION METHOD

Defining classes of objects with deformable templates consists in describing all the elements from a class with a single template. Each model will define a class of object and an instance of an object will be defined by its description given by the model. So each deformable template stores the variability between different instances of the same class object.

Each cluster of shapes is associated with a deformable template model. This model contains a prototype, that is, the the point that best represents the class, and the cluster limits. The prototype is given by the mean shape x_m , and the cluster limits are defined by the axes of deformation (the eigenvectors stored in \tilde{P}) and the upper bound of deformation T^2 . The model is built during a training phase, as described in section 2.1.

The diagram in figure 4 shows two such deformable template models, projected on the 2D space formed by their main two eigenvectors. The eigenvectors in \tilde{P} define the main axes of the two clusters. Since by construction the two eigenvectors are orthogonal, the boundary of the cluster is an ellipse. The radius of the ellipse is given by T^2 . If the confidence interval $\alpha = 0.95$ is chosen, the boundary will contain 95% of shapes in the cluster.

Using PDM-based deformable templates, classification consists in determining, for a given shape, which model produces the smallest fitting error defined in equations (10) and (21). Because fitting is optimization procedure that can be lengthy, a 2-pass strategy is used to speed up classification¹². A first pass measures the euclidean distance to the prototypes. The classes with the smallest square error are selected and a constrained fitting procedure is applied exclusively on their models. The shape is classified as the class whose model produces the least error.

Figure 4 shows schematically the advantage gained by introducing the deformation constraint into PDMs. The shape drawn by a black square happens to be closer to the prototype in cluster B than to that in cluster A. Without constraining the deformation, the shape would have been assigned to class B, the closest one. Even though it would have produced a very deformed shape B. When constraining the template fitting process, cluster A yields the smallest error.

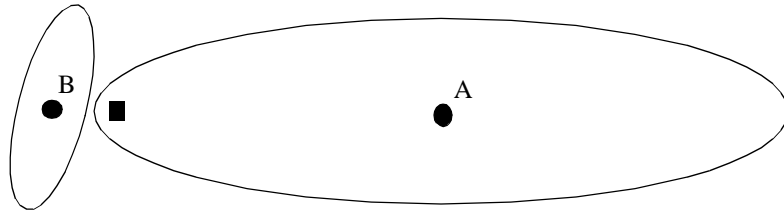


Figure 4: Representation of two clusters in two dimensions. The black circles represent the prototype of the cluster and the square is the shape to be classified.

4.RESULTS

For training and testing the system different images of crystal stones have been collected with a camera. The images have been binarized, to make easier the edge detection (cf. Figure 5a). A hundred of images have been used for each training set and a set of 20 images have been kept for the tests.

The model works with a hundred of points lying on the contour of the shape, that means working with 200 variables since for each point we have two coordinates x and y.

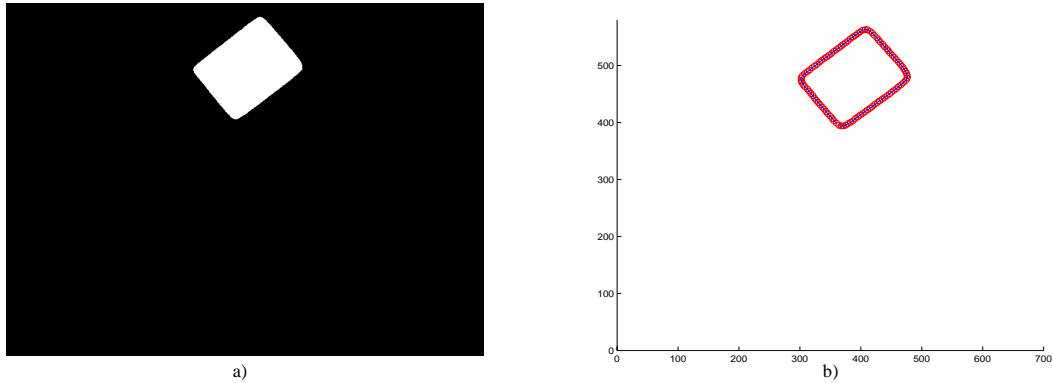


Figure 5: a) A binarized image taken with the camera. b) Shapepoints extracted from the contour.

4.1 Template fitting and classification

The system is tested with different samples to verify its right performance with the rest of images that have not been used during the training phase. A confusion matrix has been built with the results of this test:

Table 1: Confusion matrix

	1	2	3	Defective	Total	Success	Error
1	20				20	100.0%	0.0%
2		20			20	100.0%	0.0%
3			20		20	100.0%	0.0%
Defective				20	20	100.0%	0.0%
Total	20	20	20	20	80	100.0%	0.0%

As it can be seen from the confusion matrix all the samples have been correctly classified. The shapes appeared randomly in any orientation and position.

In figure Figure 6 it is shown how the model tries to fit the contour of different shapes. When the shape presents an unacceptable deformation the model gives the best approximation but keeping the deformable parameters inside the limits of the constraint.

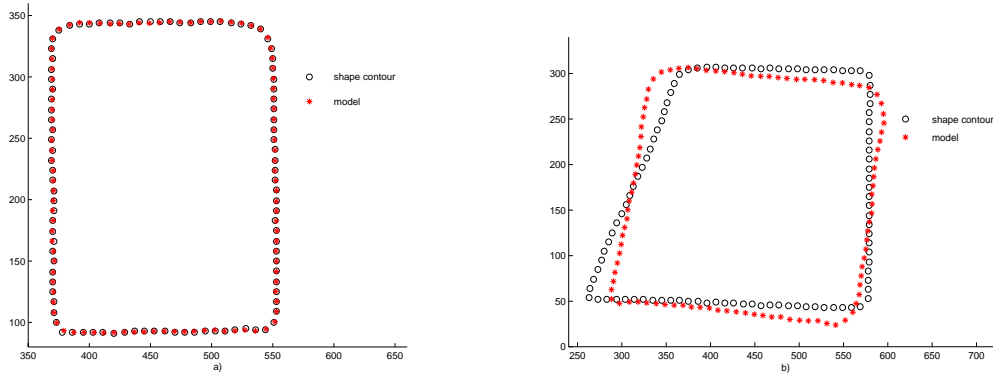


Figure 6: a) The model has fitted a correct shape. b) Best approximation to the shape restricted to the constraint T^2 .

Figure 7 shows the final result of the fitting process with and without constraint. In the first case the shape has been rejected in the second has been classified as belonging to class 2.

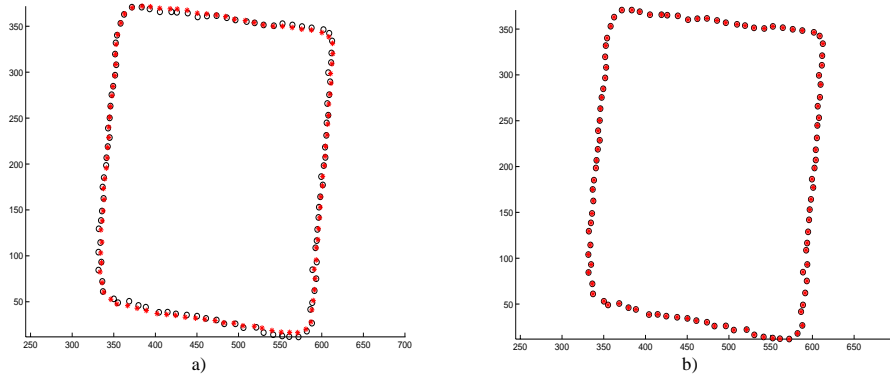


Figure 7: a) Model with the constraint T^2 . b) Unconstrained model.

5.CONCLUSIONS

A system based on deformable templates has been developed for the analysis and classification of objects according to their shape. The main contribution is the addition of a constraint to the Point Distribution Model based on Hotelling's T^2 .

The results show that we can match different objects independently of their location, orientation and size. The system accepts some deformation on the contour of the shape but restricting the directions of deformation and the amount of deformation in each direction based on the results of the training phase. This restrictions allows to detect deformations that exceed the specifications based on the well-known and widely used statistic as T^2 . From experimental results, one can see that, in some cases, the model without the constraint would have classified parts that did not fit inside the specifications and with the addition of the constraint they can be correctly rejected. This constraint affords us to fix the limits

of the clusters of the different parts, so the classification is not done according to the distance to a prototype but looking if the part falls inside the limits of a cluster.

Furthermore, although not shown in this paper, the use of Hotelling's T^2 chart opens a direct way to evaluate the performance of the manufacturing process under inspection. Indeed data collected during the process of shape analysis and classification can be used to compute multivariate Process Capability Indices²⁰.

REFERENCES

1. D. M. Gavril, J. Giebel, H. Neumann, "Learning Models from Examples", B. Radig and S. Florczyk (Eds.): *DAGM 2001*, LNCS 2191, pp. 369-376, 2001.
2. Jinhai Cai, Zhi-Qiang Liu, "Hidden Markov models with spectral features for 2D shape recognition", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol: 23 Issue: 12, pp. 1454 -1458, Dec. 2001.
3. K. Nakamura, S. Miyamoto and T. Yoshikawa, "Rotation, size and shape recognition by a spreading neural network", *Neural Information Processing*, 1999. Proceedings. ICONIP '99. 6th International Conference on, Vol: 3, pp. 998 -1003, 1999.
4. T. Singh, R. Krishnan, R.P. Arora, "Feature based shape recognition using Hopfield neural network", *Industrial Automation and Control*, 1995 (I A & C'95), IEEE/IAS International Conference on (Cat. No.95TH8005), pp: 19 -24, 1995.
5. M. Suing, L. Hermes, J. M. Buhmann, "A New Contour-Based Approach to Object Recognition for Assembly Line Robots", B. Radig and S. Florczyk (Eds.): *DAGM 2001*, LNCS 2191, pp. 329-336, 2001.
6. A. K. Jain, Y. Zhong, S. Lakshmanan, "Object Matching Using Deformable Templates", *IEEE Transactions on pattern analysis and machine intelligence*, Vol.18, No 3, March 1996.
7. T. Zimmerman, *Capturing a-priori knowledge by training two-dimensional deformable templates*, Ph.D. thesis, EPFL, 2002.
8. Visual Automation Ltd., *Active Shape Model Toolkit*, <http://www.isbe.man.ac.uk/val/asmtk/index.html>.
9. A. A. Al-Shaher, E. R. Hancock, "Linear Shape Recognition with Mixtures of Point distribution Models", T. Caelli et al. (Eds.): *SSPR&SPR 2002*, LNCS 2396, pp. 205-215, 2002.
10. T. F. Cootes, D. H. Cooper, and J. Graham, "Training models of shapes from sets of examples". In *British Machine Vision Conference 92*, pages 8-18. Springer-Verlag, 1992.
11. T. F. Cootes, C. J. Taylor, D. H. Cooper, and J. Graham. "Active shape models - their training and applications". *Computer Vision and Image Understanding*, 61(1):38-59, 1995.
12. T. Aïxut, *Shape recognition using deformable templates*, Project Report, LPM-IPR-EPFL, Lausanne, 2002.
13. J. Edward Jackson, "Principal Components and Factor Analysis Part I - Principal Components", *Journal of Quality Technology*, Vol. 12, No 4, October 1980, 201-213.
14. R. L. Mason, Youn-Min Chou, J. C. Young, "Applying Hotelling's T^2 Statistic to Batch Processes", *Journal of Quality Technology*, Vol 33, No. 4, October 2001.
15. B. Jorgensen, J. Rajeswaran, *A generalization of Hotelling's T^2* , Research report: Department of Statistics and Demography, University of Southern Denmark, March 1999.
16. L. Vandenberghe, S. Boyd, "Semidefinite programming", *SIAM Review*, Vol. 38, No. 1, pp. 49-95, March 1996.
17. T. F. Cootes and C. J. Taylor, "Active shape models: 'smart snakes'". In *British Machine Vision Conference 92*, pages 266-275. Springer-Verlag, 1992.
18. A. K. Jain, *Fundamentals of digital image processing*, Prentice Hall, 1989.
19. C. Bauckhage, J. Fritsch, G. Sagerer, "Memorizing Visual Knowledge for Assembly Process Monitoring", B. Radig and S. Florczyk (Eds.): *DAGM 2001*, LNCS 2191, pp. 178-185, 2001.
20. Douglas C. Montgomery, *Introduction to statistical quality control*, third edition, John Wiley & Sons, Inc., 1996.